# Exploratory Testing Session 2 Report

**ECSE 429 – Software Validation Term Project (Part A)**

## 1. Session Information

**Session ID:** Exploratory Session 2
**Date:** 18/01/25
**Duration:** 45 minutes
**Participants:**

- Name: David Zhou
- Student ID: 261135446
- Email: david.zhou3@mail.mcgill.ca

**Application Under Test:** Thingifier REST API Todo List Manager v1.5.2 - /todos endpoint
**Environment:** Localhost (`http://localhost:4567`)
**Tools Used:** Terminal (Ghostty), curl, browser (Chrome)

## 2. Charter

Identify capabilities and areas of potential instability of the "REST API Todo List Manager".
Identify documented and undocumented "REST API Todo List Manager" capabilities.
For each capability, create scripts to demonstrate the capability.
Exercise each capability identified with data typical to the intended use of the application.

## 3. Session Notes

The script for the following session notes can be found in p1/scripts/session2_todos_exploration.sh

| Method | Method Type | is documented? | Expected Behavior |
|---|---|---|---|
| /todos/{id}/categories | GET | DOCUMENTED | Return all categories related to the todo |
| /todos/{id}/task-of | GET | DOCUMENTED | Return all the projects related to the todo |
| /todos/{id}/categories | POST | DOCUMENTED | Create relationship using category id |
| /todos/{id}/task-of | POST | DOCUMENTED | Create relationship using project id |

| | | | |
|---|---|---|---|
| /todos/:id/categories/{id} | DELETE | DOCUMENTED | Remove category-todo relationship |
| /todos/{id}/task-of/{id} | DELETE | DOCUMENTED | Remove project-todo relationship |
| /todos/{id} | PATCH | UNDOCUMENTED | Reject unsupported method |
| /todos | TRACE | UNDOCUMENTED | Reject unsupported method |
| /todos (XML header + JSON body) | POST | UNDOCUMENTED | Reject mismatched format |
| /todos (no content-type) | POST | UNDOCUMENTED | Reject or default safely |

| expected == observed | Observed Behavior | Side Notes | XML & Screenshots |
|---|---|---|---|
| Yes | /todos/1/categories returned linked categories | Confirms relationship sub-resource exists | P1/screenshots/session2-1.png |
| Yes | /todos/1/task-of returned linked projects | Confirms project linkage | P1/screenshots/session2-1.png |
| No | Relationship POST rejected ID usage | Documentation implies POST should work | P1/screenshots/session2-1.png |
| Yes | Relationship DELETE removed links silently | Nor response body returned | P1/screenshots/session2-1.png |
| No | PATCH returned HTML 404 page | Inconsistent with JSON API | P1/screenshots/session2-1.png |
| No | TRACE returned HTML error page | Framework -level response | P1/screenshots/session2-1.png |
| No | JSON accepted with XML content-type | Content-type ignored | P1/screenshots/session2-1.png |
| No | JSON accepted without content-type | Header not validated | P1/screenshots/session2-1.png |
| No | XML with JSON header leaked Java | Internal implementation | P1/screenshots/session2-1.png |

| | | | |
|---|---|---|---|
| | exception | exposed | |

## 4. Summary of Session Findings

- The `/todos` resource exposes extended relationship endpoints that allow retrieval of related categories and projects. These relationships are prepopulated with test data on startup.
- Attempts to create relationships using POST with an ID were rejected, contradicting the documented behavior of the API. However, relationship deletion endpoints functioned correctly, although they returned no confirmation response.
- Several undocumented behaviors were identified. The API accepts JSON requests even when the `Content-Type` header is set incorrectly or omitted entirely. Unsupported HTTP methods such as PATCH and TRACE returned HTML error pages rather than JSON-formatted errors. Submitting XML data with a JSON content type resulted in internal Java parsing exceptions being exposed to the client.
- These results indicate weaknesses in protocol validation, documentation accuracy, and error-handling consistency.

## 5. Summary of Concerns

- The API does not validate `Content-Type` headers.
- Java exception messages are exposed to clients.
- Relationship creation behavior contradicts documentation.
- Unsupported methods return HTML instead of structured API responses.
- Relationship deletion operations return no confirmation data.
- Error-handling behavior is inconsistent across endpoints.

## 6. Test Ideas

- Header fuzzing and malformed content-type testing
- Security testing on unsupported HTTP methods
- Cardinality testing of todo relationships
- Payload size and encoding stress tests
- Verification of error format consistency
- Rapid relationship creation/deletion stability testing